AF/2183

PTO/SB/21 (09-04)
Approved for use through 07/31/2006. OMB 0651-0031
U.S. Patent and Trademark Office; U.S. DEPARTMENT OF COMMERCE

TRANSMITTAL FORM

(to be used for all correspondence after initial filing)

Total Number of Pages in This Submission	18

Application Number	09/548,469
Filing Date	April 13, 2000
First Named Inventor	Balaram Sinharoy et al.
Art Unit	2183
Examiner Name	Aimee J. Li
Attorney Docket Number	AT9-99-129

ENCLOSURES *(Check all that apply)*

- ☐ Fee Transmittal Form
  - ☐ Fee Attached
- ☐ Amendment/Reply
  - ☐ After Final
  - ☐ Affidavits/declaration(s)
- ☐ Extension of Time Request
- ☐ Express Abandonment Request
- ☐ Information Disclosure Statement
- ☐ Certified Copy of Priority Document(s)
- ☐ Reply to Missing Parts/ Incomplete Application
  - ☐ Reply to Missing Parts under 37 CFR 1.52 or 1.53
- ☐ Drawing(s)
- ☐ Licensing-related Papers
- ☐ Petition
- ☐ Petition to Convert to a Provisional Application
- ☐ Power of Attorney, Revocation Change of Correspondence Address
- ☐ Terminal Disclaimer
- ☐ Request for Refund
- ☐ CD, Number of CD(s) _____
  - ☐ Landscape Table on CD
- ☐ After Allowance Communication to TC
- ☐ Appeal Communication to Board of Appeals and Interferences
- ☑ Appeal Communication to TC **(Appeal Notice, Brief, Reply Brief)**
- ☐ Proprietary Information
- ☐ Status Letter
- ☑ Other Enclosure(s) (please Identify below):

Request for Reinstatement of Appeal

Remarks

Supplemental Appeal Brief

SIGNATURE OF APPLICANT, ATTORNEY, OR AGENT

Firm Name	Winstead Sechrest & Minick P.C.		
Signature			
Printed name	Robert A. Voigt, Jr.		
Date	December 9, 2004	Reg. No.	47,159

CERTIFICATE OF TRANSMISSION/MAILING

I hereby certify that this correspondence is being facsimile transmitted to the USPTO or deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to: Mail Stop Appeal Brief- Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450 on the date shown below:

Signature	Serena Beller		
Typed or printed name	Serena Beller	Date	December 9, 2004

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re Application of: Balaram Sinharoy et al.	:	Before the Examiner: Li, Aimee J.
Serial No.: 09/548,469	:	Group Art Unit: 2183
Filed: April 13, 2000	:	
Title: USE OF SOFTWARE HINT FOR BRANCH PREDICTION IN THE ABSENCE OF HINT BIT IN THE BRANCH INSTRUCTION	:	IBM Corporation Intellectual Property Law 11400 Burnet Road Austin, Texas 78758

REQUEST FOR REINSTATEMENT OF APPEAL

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

In response to the Office Action (Paper No. 13) having a mailing date of November 5, 2004, reopening prosecution of the above-referenced Application, Applicants respectfully request reinstatement of the Appeal based on the Appeal Brief filed on August 9, 2004 and the Notice of Appeal filed on June 9, 2004.

CERTIFICATION UNDER 37 C.F.R. §1.8

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to Mail Stop Appeal Brief-Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on December 9, 2004.

Serena Beller
Signature

Serena Beller
(Printed name of person certifying)

A supplemental appeal brief is filed herewith.

FEE DEFICIENCY

NOTE: If there is a fee deficiency and there is no authorization to charge an account, additional fees are necessary to cover the additional time consumed in making up the original deficiency. If the maximum, six-month period has expired before the deficiency is noted and corrected, the application is held abandoned. In those instances where authorization to charge is included, processing delays are encountered in returning the papers to the PTO Finance Branch in order to apply these charges prior to action on the cases. Authorization to charge the deposit account for any fee deficiency should be checked. See the Notice of April 7, 1986, 1065 O.G. 31-33.

☒ If any additional extension and/or fee is required, this is a request therefor and to charge Account No. 09-0447 (AT9-99-129).

AND/OR

☒ If any additional fee for claims is required, charge Account No. 09-0447 (AT9-99-129).

Respectfully submitted,

WINSTEAD SECHREST & MINICK P.C.

Attorneys for Applicants

By: _____

Kelly K. Kordzik
Reg. No. 36,571
Robert A. Voigt, Jr.
Reg. No. 47,159

P.O. Box 50784
Dallas, Texas 75201
(512) 370-2832

BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES

In re Application of: Balaram Sinharoy et al.	:	Before the Examiner: Li, Aimee J.
Serial No.: 09/548,469	:	Group Art Unit: 2183
Filed: April 13, 2000	:	
	:	IBM Corporation
Title: USE OF SOFTWARE HINT FOR BRANCH PREDICTION IN THE ABSENCE OF HINT BIT IN THE BRANCH INSTRUCTION	:	Intellectual Property Law 11400 Burnet Road Austin, Texas 78758

SUPPLEMENTAL APPEAL BRIEF

Mail Stop Appeal Brief-Patents
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Dear Sir:

This supplemental brief is being submitted pursuant to 37 C.F.R. §41.37.

CERTIFICATION UNDER 37 C.F.R. §1.8

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as first class mail in an envelope addressed to Mail Stop Appeal Brief-Patents, Commissioner for Patents, P.O. Box 1450, Alexandria, Virginia 22313-1450, on December 9, 2004.

Serena Beller
Signature

Serena Beller
(Printed name of person certifying)

I. INCORPORATION BY REFERENCE

Appellants hereby incorporate herein by reference Sections I-V and VIII-IX of Appellants' Brief mailed on August 9, 2004.

II. GROUNDS OF REJECTION TO BE REVIEWED ON APPEAL

Claims 39 and 40 stand rejected under 35 U.S.C. §102(b) as being anticipated by Hennessy's Computer Architecture: A Quantitative Approach Second Edition © 1996 (hereinafter "Hennessy"). Claims 1-14 stand rejected under 35 U.S.C. §103(a) as being unpatentable over Henry et al. (U.S. Patent No. 6,550,004) (hereinafter "Henry") in view of Tanenbaum's Structured Computer Organization Second Edition © 1984 (hereinafter "Tanenbaum").

III. ADDITIONAL ARGUMENTS

A. Claims 39 and 40 are not properly rejected under 35 U.S.C. §102(b).

The Examiner has rejected claims 39 and 40 under 35 U.S.C. §102(b) as being anticipated by Hennessy. Paper No. 13, page 2. Appellants respectfully traverse and assert that claims 39 and 40 are not properly rejected under 35 U.S.C. §102(b) as being anticipated by Hennessy for at least the reasons stated below.

For a claim to be anticipated under 35 U.S.C. §102, each and every claim limitation must be found within the cited prior art reference and arranged as required by the claim. M.P.E.P. §2131.

Appellants respectfully assert that Hennessy does not disclose "determining if the conditional branch instruction is positioned at a specified address in a sequence of instructions being executed" as recited in claim 39. The Examiner cites page 273, paragraph 2; Figure 4.22; page 274, paragraph 1; and Figure 4.23 of Hennessy as disclosing the above-cited claim limitation. Paper No. 13, page 3. Appellants respectfully traverse and assert that Hennessy instead discloses that the PC of the

instruction being fetched is matched against a set of instruction addresses stored in the first column of a buffer. Hennessy further discloses that these instruction addresses represent the addresses of known branches. Hennessy further discloses that if the PC matches one of these entries, then the instruction being fetched is a taken branch, and the second column in the buffer, predicted PC, contains the prediction for the next PC after the branch. Hennessy further discloses that fetching begins immediately at that address. Hence, Hennessy discloses matching the instruction fetched with a set of stored instruction addresses. This is not the same as determining if an instruction is positioned at a specified address in a sequence of instructions. Neither is there language in the cited passages and figures that discloses determining if a conditional branch instruction is positioned at a specified address in a sequence of instructions being executed. Thus, Hennessy does not disclose all of the limitations of claim 39, and thus Hennessy does not anticipate claim 39. M.P.E.P. §2131.

Appellants further assert that Hennessy does not disclose "predicting whether the conditional branch instruction will be taken or not taken as a function of the position of the specified address" as recited in claim 39. The Examiner cites page 273, paragraph 2; Figure 4.22; page 274, paragraph 1; and Figure 4.23 of Hennessy as disclosing the above-cited claim limitation. Paper No. 13, page 3. Appellants respectfully traverse. As stated above, Hennessy instead discloses that the PC of the instruction being fetched is matched against a set of instruction addresses stored in the first column of a buffer. Hennessy further discloses that these instruction addresses represent the addresses of known branches. Hennessy further discloses that if the PC matches one of these entries, then the instruction being fetched is a taken branch, and the second column in the buffer, predicted PC, contains the prediction for the next PC after the branch. Hennessy further discloses that fetching begins immediately at that address. Hence, Hennessy discloses matching the instruction fetched with a set of stored instruction addresses. This is not the same as predicting whether a conditional branch instruction will be taken or not taken as a function of the position of the specified address. Thus, Hennessy does not disclose all of the

limitations of claim 39, and thus Hennessy does not anticipate claim 39. M.P.E.P. §2131.

Claim 40 recites combinations of features including the above combinations, and thus is not anticipated for at least the above-stated reasons. Claim 40 recites additional features which, in combination with the features of the claim upon which it depends, is not anticipated by Hennessy.

For example, Hennessy does not disclose "wherein the predicting program step will predict taken if the specified address is a multiple of specified number N" as recited in claim 40. The Examiner cites page 273, paragraph 2; Figure 4.22; page 274, paragraph 1; and Figure 4.23 of Hennessy as disclosing the above-cited claim limitation. Paper No. 13, page 3. The Examiner further states:

> In regards to Hennessy, every address is a multiple of itself. For example, if the branch instruction is at specified address 20, then the address is a multiple of a specified number 20. Paper No. 13, page 3.

Appellants respectfully traverse. As stated above, Hennessy instead discloses that the PC of the instruction being fetched is matched against a set of instruction addresses stored in the first column of a buffer. Hennessy further discloses that these instruction addresses represent the addresses of known branches. Hennessy further discloses that if the PC matches one of these entries, then the instruction being fetched is a taken branch, and the second column in the buffer, predicted PC, contains the prediction for the next PC after the branch. Hennessy further discloses that fetching begins immediately at that address. Hence, Hennessy discloses matching the instruction fetched with a set of stored instruction addresses. This is not the same as predicting taken if the specified address in the sequence of instructions being executed is a multiple of specified number N. Thus, Hennessy does not disclose all of the limitations of claim 40, and thus Hennessy does not anticipate claim 40. M.P.E.P. §2131.

Furthermore, as stated above, Hennessy does not disclose determining if a conditional branch instruction is positioned at a specified address for at least the reasons stated above. Hence, Hennessy does not disclose predicting taken if the specified address is a multiple of specified number N. Thus, Hennessy does not disclose all of the limitations of claim 40, and thus Hennessy does not anticipate claim 40. M.P.E.P. §2131.

Furthermore, the Examiner's example does not relate to predicting the conditional branch instruction taken if the specified address in the sequence of instructions being executed is a multiple of specified number N. Hence, Hennessy does not disclose all of the limitations of claim 40, and thus Hennessy does not anticipate claim 40. M.P.E.P. §2131.

As a result of the foregoing, Appellants respectfully assert that not each and every claim limitation was found within Hennessy, and thus claims 39-40 are not anticipated by Hennessy.

B. Claims 1-14 are not properly rejected under 35 U.S.C. §103(a) as being unpatentable over Henry in view of Tanenbaum.

1. The Examiner has not provided any objective evidence for modifying Henry to provide a software branch prediction.

A *prima facie* showing of obviousness requires the Examiner to establish, *inter alia*, that the prior art references teach or suggest, either alone or in combination, all of the limitations of the claimed invention, and the Examiner must provide a motivation or suggestion to combine or modify the prior art reference to make the claimed inventions. M.P.E.P. §2142. The showings must be clear and particular and supported by objective evidence. *In re Lee*, 277 F.3d 1338, 1343, 61 U.S.P.Q.2d 1430, 1433-34 (Fed. Cir. 2002); *In re Kotzab*, 217 F.3d 1365, 1370, 55 U.S.P.Q.2d 1313, 1317 (Fed. Cir. 2000); *In re Dembiczak*, 50 U.S.P.Q.2d. 1614,

1617 (Fed. Cir. 1999). Broad conclusory statements regarding the teaching of multiple references, standing alone, are not evidence. *Id.*

The Examiner's motivation for modifying Henry with Tanenbaum to provide a software branch prediction of the conditional branch instruction as a function of the determination if the specified condition register field is used to store the branch condition of the conditional branch instruction, as recited in claim 1 and similarly in claim 8, is that "the branch prediction method taught in Henry can be done in both hardware and software and it is more a design decision whether to implement the method in hardware and software." Paper No. 13, pages 5 and 6. The Examiner's motivation is insufficient to support a *prima facie* case of obviousness for at least the reasons stated below.

The Examiner's motivation does not address as to why one of ordinary skill in the art would modify Henry's branch predictor implemented in hardware to be implemented in software. While an instruction executed by hardware can be simulated in software, the Examiner has not provided motivation for actually modifying Henry's branch predictor implemented in hardware to be implemented in software. The mere fact that a reference can be modified does not render the resultant combination obvious unless the prior art also suggests the desirability of the modification. *In re Mills*, 916 F.2d 680, 16 U.S.P.Q.2d 1430 (Fed. Cir. 1990); M.P.E.P. §2143.01. The Examiner is merely relying upon her own subjective opinion which is insufficient to support a *prima facie* case of obviousness. *In re Lee*, 61 U.S.P.Q.2d 1430, 1434 (Fed. Cir. 2002). Consequently, the Examiner's motivation is insufficient to support a *prima facie* case of obviousness for rejecting claims 1-14. *Id.*

2. The Examiner has not presented a reasonable expectation of success when modifying Henry to provide a software branch prediction.

The Examiner must present a reasonable expectation of success in modifying Henry's branch predictor implemented in hardware to be implemented in software (Examiner admits that Henry does not teach performing branch prediction by software) in order to establish a *prima facie* case of obviousness. *In re Merck & Co., Inc.*, 800 F.2d 1091, 231 U.S.P.Q. 375 (Fed. Cir. 1986); M.P.E.P. §2143.02.

Henry teaches a branch predictor that includes global and local Agree dynamical branch predictors, one of which is selected for correlation with a static branch prediction made based upon a test type of a conditional branch instruction specifying a condition upon which the branch will be taken. Abstract. Henry further teaches that the selection is made by correlating a selection prediction made by the static predictor based on the test type and an Agree prediction made by a selector history table based on the branch instruction address. Abstract. Henry further teaches that the dynamic predictors are updated if they are selected and incorrectly predicted the outcome. Abstract. Henry further teaches that the selector history table is updated if the selected dynamic predictor predicted incorrectly and the non-selected dynamic predictor predicted correctly. Abstract. Henry further teaches a hardware mechanism, as illustrated in Figure 2, for implementing the branch prediction as described above.

The Examiner has not provided any evidence as to how Henry's hardware mechanism, as illustrated in Figure 2, would be modified to be implemented in software or the feasibility of implementing the functionality of Henry's hardware mechanism in software. The Examiner must provide objective evidence as to how Henry's hardware mechanism, as illustrated in Figure 2, would be modified to be implemented in software or the feasibility of implementing the functionality of Henry's hardware mechanism in software. M.P.E.P. §2143.02. Since the Examiner has not provided such evidence, the Examiner has not presented a reasonable expectation of success in modifying Henry's branch predictor implemented in

hardware to be implemented in software in order to establish a *prima facie* case of obviousness. *In re Merck & Co., Inc.*, 800 F.2d 1091, 231 U.S.P.Q. 375 (Fed. Cir. 1986); M.P.E.P. §2143.02. Accordingly, the Examiner has not presented a *prima facie* case of obviousness in rejecting claims 1-14. M.P.E.P. §2143.02.

3. Henry and Tanenbaum, taken singly or in combination, do not teach or suggest the following claim limitations.

Appellants respectfully assert that Henry and Tanenbaum, taken singly or in combination, do not teach or suggest "providing a software branch prediction of the conditional branch instruction as a function of the determination if the specified condition register field is used to store the branch condition of the conditional branch condition" as recited in claim 1 and similarly in claim 8. The Examiner cites lines 13-14 of the Abstract; column 4, lines 49-52; column 5, lines 8-12 and 35-38; column 9, lines 31-44 and Figure 2 of Henry as disclosing the above-cited claim limitation. Paper No. 13, page 4. Appellants respectfully traverse and assert that Henry instead teaches a static predictor, static predictor 222, that receives three inputs and predicts the outcome of conditional branch instructions based upon the three inputs. Henry further teaches that one of the three inputs comprises a conditional branch instruction test type. Henry further teaches that the test type specifies a condition upon which the branch instruction will be taken or not taken. Henry further teaches that the test type includes x86 conditional jump instruction (JCC) test types. Henry further teaches that the x86 conditional jump instruction test types include conditionsbased upon the carry, overflow, zero, parity and sign flags of the x86 FLAGS register. The Examiner cites Figure 3-9 and Table 3-2 on page 3-14 of the Intel Pentium™ Processor Family Developer's Manual Volume 3 (hereinafter "Intel") which teaches that conditional jumps and subroutine calls allow a program to sense the state of the status flags and respond to them. Paper No. 13, page 8. However, responding to the state of a flag, e.g., the state of a flag (0 or logical value of 1) may be used to suppress a conditional jump, is not the same as providing a prediction based on whether a specified field,

e.g., field 1, is used to store a condition. Therefore, the Examiner has not presented a *prima facie* case of obviousness in rejecting claims 1 and 8, since the Examiner is relying upon an incorrect, factual predicate in support of the rejection. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1455 (Fed. Cir. 1998).

Claims 2-7 and 9-14 recite combinations of features including the above combinations, and thus are patentable for at least the above-stated reasons. Claims 2-7 and 9-14 recites additional features which, in combination with the features of the claims upon which they depend, are patentable over Henry in view of Tanenbaum.

For example, Henry and Tanenbaum, taken singly or in combination, do not teach or suggest "wherein the software branch prediction predicts that the conditional branch instruction will be taken if the specified condition register field is used to store the branch condition of the conditional branch instruction" as recited in claim 2 and similarly in claim 9. The Examiner cites lines 13-14 of the abstract; column 4, lines 49-52; column 5, lines 8-12 and 35-38; column 9, lines 31-44 and Figure 2 of Henry as disclosing the above-cited claim limitation. Paper No. 13, page 6. Appellants respectfully traverse. As stated above, Henry instead teaches a static predictor, static predictor 222, that receives three inputs and predicts the outcome of conditional branch instructions based upon the three inputs. Henry further teaches that one of the three inputs comprises a conditional branch instruction test type. Henry further teaches that the test type specifies a condition upon which the branch instruction will be taken or not taken. Henry further teaches that the test type includes x86 conditional jump instruction (JCC) test types. Henry further teaches that the x86 conditional jump instruction test types include conditionsbased upon the carry, overflow, zero, parity and sign flags of the x86 FLAGS register. The Examiner cites Figure 3-9 and Table 3-2 on page 3-14 of Intel which teaches that conditional jumps and subroutine calls allow a program to sense the state of the status flags and respond to them. Paper No. 13, page 8. However, responding to the state of a flag, e.g., the state of a flag (0 or logical value of 1) may be used to suppress a conditional jump, is

not the same as providing a prediction based on whether a specified field, e.g., field 1, is used to store a condition. Hence, Henry does not teach making a branch prediction based on whether a particular condition register field is used to store a branch condition. Neither does Henry teach predicting that a conditional branch instruction will be taken if the specified condition register field is used to store the branch condition of the conditional branch instruction. Therefore, the Examiner has not presented a *prima facie* case of obviousness in rejecting claims 2 and 9, since the Examiner is relying upon an incorrect, factual predicate in support of the rejection. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1455 (Fed. Cir. 1998).

Appellants further assert that Henry and Tanenbaum, taken singly or in combination, do not teach or suggest "wherein the software branch prediction predicts that the conditional branch instruction will be not taken if the specified condition register field is not used to store the branch condition of the conditional branch instruction" as recited in claim 3 and similarly in claim 10. The Examiner cites lines 13-14 of the abstract; column 4, lines 49-52; column 5, lines 8-12 and 35-38; column 9, lines 31-44 and Figure 2 of Henry as disclosing the above-cited claim limitation. Paper No. 13, page 6. Appellants respectfully traverse. As stated above, Henry instead teaches a static predictor, static predictor 222, that receives three inputs and predicts the outcome of conditional branch instructions based upon the three inputs. Henry further teaches that one of the three inputs comprises a conditional branch instruction test type. Henry further teaches that the test type specifies a condition upon which the branch instruction will be taken or not taken. Henry further teaches that the test type includes x86 conditional jump instruction (JCC) test types. Henry further teaches that the x86 conditional jump instruction test types include conditionsbased upon the carry, overflow, zero, parity and sign flags of the x86 FLAGS register. The Examiner cites Figure 3-9 and Table 3-2 on page 3-14 of Intel which teaches that conditional jumps and subroutine calls allow a program to sense the state of the status flags and respond to them. Paper No. 13, page 8. However, responding to the state of a flag, e.g., the state of a flag (0 or logical value of 1) may

be used to suppress a conditional jump, is not the same as providing a prediction based on whether a specified field, e.g., field 1, is not used to store a condition. Hence, Henry does not teach making a branch prediction based on whether a particular condition register field is not used to store a branch condition. Neither does Henry teach predicting that a conditional branch instruction will not be taken if the specified condition register field is not used to store the branch condition of the conditional branch instruction. Therefore, the Examiner has not presented a *prima facie* case of obviousness in rejecting claims 3 and 10, since the Examiner is relying upon an incorrect, factual predicate in support of the rejection. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1455 (Fed. Cir. 1998).

Appellants further assert that Henry and Tanenbaum, taken singly or in combination, do not teach or suggest "wherein the software branch prediction predicts that the conditional branch instruction will be not taken if the specified condition register field is used to store the branch condition of the conditional branch instruction" as recited in claim 4 and similarly in claim 11. The Examiner cites lines 13-14 of the abstract; column 4, lines 49-52; column 5, lines 8-12 and 35-38; column 9, lines 31-44 and Figure 2 of Henry as disclosing the above-cited claim limitation. Paper No. 13, page 7. Appellants respectfully traverse. As stated above, Henry instead teaches a static predictor, static predictor 222, that receives three inputs and predicts the outcome of conditional branch instructions based upon the three inputs. Henry further teaches that one of the three inputs comprises a conditional branch instruction test type. Henry further teaches that the test type specifies a condition upon which the branch instruction will be taken or not taken. Henry further teaches that the test type includes x86 conditional jump instruction (JCC) test types. Henry further teaches that the x86 conditional jump instruction test types include conditionsbased upon the carry, overflow, zero, parity and sign flags of the x86 FLAGS register. The Examiner cites Figure 3-9 and Table 3-2 on page 3-14 of Intel which teaches that conditional jumps and subroutine calls allow a program to sense the state of the status flags and respond to them. Paper No. 13, page 8. However,

responding to the state of a flag, e.g., the state of a flag (0 or logical value of 1) may be used to suppress a conditional jump, is not the same as providing a prediction based on whether a specified field, e.g., field 1, is used to store a condition. Hence, Henry does not teach making a branch prediction based on whether a particular condition register field is used to store a branch condition. Neither does Henry teach predicting that a conditional branch instruction will not be taken if the specified condition register field is used to store the branch condition of the conditional branch instruction. Therefore, the Examiner has not presented a *prima facie* case of obviousness in rejecting claims 4 and 11, since the Examiner is relying upon an incorrect, factual predicate in support of the rejection. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1455 (Fed. Cir. 1998).

Appellants further assert that Henry and Tanenbaum, taken singly or in combination, do not teach or suggest "wherein the software branch prediction predicts that the conditional branch instruction will be taken if the specified condition register field is not used to store the branch condition of the conditional branch instruction" as recited in claim 5 and similarly in claim 12. The Examiner cites lines 13-14 of the abstract; column 4, lines 49-52; column 5, lines 8-12 and 35-38; column 9, lines 31-44 and Figure 2 of Henry as disclosing the above-cited claim limitation. Paper No. 13, page 7. Appellants respectfully traverse. As stated above, Henry instead teaches a static predictor, static predictor 222, that receives three inputs and predicts the outcome of conditional branch instructions based upon the three inputs. Henry further teaches that one of the three inputs comprises a conditional branch instruction test type. Henry further teaches that the test type specifies a condition upon which the branch instruction will be taken or not taken. Henry further teaches that the test type includes x86 conditional jump instruction (JCC) test types. Henry further teaches that the x86 conditional jump instruction test types include conditionsbased upon the carry, overflow, zero, parity and sign flags of the x86 FLAGS register. The Examiner cites Figure 3-9 and Table 3-2 on page 3-14 of Intel which teaches that conditional jumps and subroutine calls allow a program to sense the state of the status flags and

respond to them. Paper No. 13, page 8. However, responding to the state of a flag, e.g., the state of a flag (0 or logical value of 1) may be used to suppress a conditional jump, is not the same as providing a prediction based on whether a specified field, e.g., field 1, is not used to store a condition. Hence, Henry does not teach making a branch prediction based on whether a particular condition register field is not used to store a branch condition. Neither does Henry teach predicting that a conditional branch instruction will be taken if the specified condition register field is not used to store the branch condition of the conditional branch instruction. Therefore, the Examiner has not presented a *prima facie* case of obviousness in rejecting claims 5 and 12, since the Examiner is relying upon an incorrect, factual predicate in support of the rejection. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1455 (Fed. Cir. 1998).

Appellants further assert that Henry and Tanenbaum, taken singly or in combination, do not teach or suggest "wherein the specified condition register field is N, where N is an integer" as recited in claim 6 and similarly in claim 13. Appellants further assert that Henry and Tanenbaum, taken singly or in combination, do not teach or suggest "wherein the specified condition register field is a multiple of N" as recited in claim 7 and similarly in claim 14. The Examiner cites column 7, lines 36-38 and column 9, lines 31-44 of Henry as teaching the above-cited claim limitation. Paper No. 13, page 7. Appellants respectfully traverse and assert that Henry instead teaches a register file, register file 105, that includes a status flags register that is used in determining whether branch conditions have been satisfied. While the status flags register is used in determining whether branch conditions have been satisfied, presumably based on the state of flags stored in such a register, Henry does not teach that a prediction is a function of whether a specified field in the status flags register was used to store a branch condition. Further, Henry does not teach that such a specified field is N or a multiple of N. Therefore, the Examiner has not presented a *prima facie* case of obviousness in rejecting claims 6, 7, 13 and 14, since the Examiner is relying upon an incorrect, factual predicate in support of the rejection. *In re Rouffet*, 47 U.S.P.Q.2d 1453, 1455 (Fed. Cir. 1998).

As a result of the foregoing, Appellants respectfully assert that there are numerous claim limitations not taught or suggested in the cited prior art, and thus the Examiner has not presented a *prima facie* case of obviousness for rejecting claims 1-14 as being unpatentable over Henry in view of Tanenbaum. M.P.E.P. §2143.

IV. CONCLUSION

For at least the reasons stated above and in the Appeal Brief filed by Appellants on August 9, 2004, the rejections of claims 1-14 and 39-40 are in error. Appellants respectfully request reversal of the rejections and allowance of claims 1-14 and 39-40.

Respectfully submitted,
WINSTEAD SECHREST & MINICK P.C.
Attorneys for Appellants

By: _____
Robert A. Voigt, Jr.
Reg. No. 47,159
Kelly K. Kordzik
Reg. No. 36,571

P.O. Box 50784
Dallas, Texas 75201
(512) 370-2832

Austin_1 267629v.1